BAB VII. FUNCTION

Dalam pemrograman, string merupakan kumpulan dari beberapa karakter-karakter. Untuk membedakan string dengan karakter, dalam C++ dibedakan penulisannya. Suatu nilai merupakan string apabila diapit dengan tanda petik ganda "...", misalnya "SAYA". Sedangkan karakter (char) diapit dengan tanda petik tunggal, misal 's'.

A. Pengantar Function

Sebuah function berisi sejumlah pernyataan yang dikemas dalam sebuah nama. Nama ini selanjutnya dapat dipanggil beberapa kali di beberapa tempat dalam program.

Tujuannya:
1. memudahkan dalam mengembangkan program. Program dibagi menjadi beberapa subprogram kecil, sehingga hal ini menjadi kunci dalam pembuatan program terstruktur.
2. menghemat ukuran program, karena beberapa perintah yang sama dan dijalankan beberapa kali dalam program dapat dijadikan satu kali saja dalam suatu function, kemudian function tersebut dapat dipanggil berulang kali.

Contoh Function I:

```
#include <iostream.h>
#include <conio.h>

void garis();          // prototype function

void main()            // main function
{
      clrscr();
      garis();         // panggil function
      cout << "NIM               NAMA MAHASISWA" << endl;
      garis();         // panggil function
      cout << "M0197001          AMIR HAMZAH " << endl;
      cout << "M0197002          PAIMAN" << endl;
      garis();         // panggil function
}

void garis()           // detail function
{
      int i;
      for(i=0;i<=40;i++)
      {
            cout << "-";
      }
      cout << endl;
}
```

Contoh di atas menggambarkan bagaimana membuat function untuk membuat garis. Nama functionnya adalah **garis**. Untuk membuat suatu function, diperlukan suatu prototype dari function tersebut. Prototype function memiliki sintaks sbb:

returned_value_data_type   nama_function(argumen);

Seperti halnya dalam Pascal, suatu function dapat mengembalikan (return) suatu nilai (value) yang tergantung tipe datanya. Tipe data value yang dikembalikan inilah yang dimaksud dengan returned_value_data_type.
Sedangkan argumen merupakan parameter-parameter yang akan diolah dalam function tersebut. Argumen boleh ada boleh tidak, sesuai kebutuhan. Apabila parameter argumennya lebih dari satu, cara penulisannya sbb:

tipe_data param1, tipe_data param2, ...

Contoh penulisan prototype function:

- double kuadrat(int x);
- float luas_segitiga(float alas, float tinggi);
- int jumlah_bil(int x, int y, int z);

Apabila suatu function tidak mengembalikan nilai, maka
returned_value_data_type nya diisi **void**.

Setelah prototype function dibuat, selanjutnya membuat function tersebut secara detail. Suatu function disebut juga subprogram, oleh karena itu strukturnya juga sama dengan struktur program utama. Pada contoh function garis() di atas, detail dari function tersebut adalah:

```
void garis()           // detail function
{
      int i;
      for(i=0;i<=40;i++)
      {
            cout << "-";
      }
      cout << endl;
}
```

Kalau diperhatikan, strukturnya sama dengan program utama **main()**.

Contoh Function II:

```
#include <iostream.h>
#include <conio.h>

float luas(float alas, float tinggi);
void main()
{
      clrscr();
      a = 10.5;
      t = 11;
      cout << "HITUNG LUAS SEGITIGA" << endl;
      cout << "Panjang alas : " << a << endl;
      cout << "Tinggi       : " << t << endl;
      cout << "Luasnya            : " << luas(a,t) << endl;
}

float luas(float alas, float tinggi)
{
```

```
      float luas_segitiga;
      luas_segitiga = alas * tinggi * 0.5;
      return luas_segitiga;
}
```

Detail function luas di atas dapat ditulis sbb:

```
float luas(float alas, float tinggi)
{
      return (alas * tinggi * 0.5);
}
```

Perintah return adalah untuk mengembalikan hasil operasi di sebelah kanannya ke perintah pemanggilan function.

B. Variabel Global dan Lokal

Setiap kali kita deklarasikan suatu variabel, belum tentu variabel tersebut dikenal di setiap function yang kita buat. Contoh:

```
#include <iostream.h>
#include <conio.h>

void cetak();

void main()
{
      int a;
      a = 10;
      cout << "Nilai a = " << a << endl;
      cetak();
}

void cetak()
{
      a++;
      cout << "Nilai a = " << a << endl;
}
```

Ketika program di atas dicompile, akan terdapat error yaitu variabel a dalam function cetak() undefined. Artinya bahwa variabel a tidak dikenal dalam cetak(). Variabel a hanya dikenal dalam program utama/ function main() saja. Maka dalam hal ini variabel a disebut variabel lokal (hanya dikenal dalam function yang di dalamnya didefinisikan a tersebut).

Selanjutnya program di atas diubah sbb:

```
#include <iostream.h>
#include <conio.h>

int a;

void cetak();
void main()
{
```

```
	a = 10;
	cout << "Nilai a = " << a << endl;
	cetak();
}

void cetak()
{
	a++;
	cout << "Nilai a = " << a << endl;
}
```

Apabila program di atas dijalankan maka akan tampil:
Nilai a = 10
Nilai a = 11

Pada program di atas, variabel a disebut variabel global karena variabel tersebut dapat dikenali di setiap function yang ada.

Bagaimana dengan yang ini???

```
#include <iostream.h>
#include <conio.h>

void cetak();
void main()
{
	int a;
	a = 10;
	cout << "Nilai a = " << a << endl;
	cetak();
}

void cetak()
{
	int a;
	cout << "Nilai a = " << a << endl;
}
```

Apabila program di atas dijalankan, hasilnya adalah:

Nilai a = 10
Nilai a = 747

Hasil di atas menunjukkan bahwa meskipun nama variabelnya sama-sama a, tapi kedua variabel a tersebut berbeda. Setiap variabel a tersebut hanya dikenali di functionnya masing-masing (tidak terkait satu dengan yang lain).

Latihan:

1. Buatlah function yang akan memberikan nilai 1 jika nilai parameter yang dimasukkan huruf ‘a’, dan akan memberikan nilai 0 jika nilai parameter yang dimasukkan selain huruf ‘a’.

2. Buatlah function untuk mencari nilai fungsi $f(x) = 2x^2 - 3x + 1$;
   Parameter functionnya adalah nilai x (tipe data float/double). Return valuenya adalah nilai f(x).

3. Buatlah function untuk membuat tampilan seperti di bawah ini:

   ```
   *
   * *
   * * *
   * *
   *
   ```

   Tampilan di atas muncul apabila dimasukkan n = 3.

   ```
   *
   * *
   * * *
   * * * *
   * * *
   * *
   *
   ```

   Tampilan di atas muncul apabila dimasukkan n = 4.

   Parameter functionnya adalah n (tipe datanya integer).

BAB VIII. STRINGS

Dalam pemrograman, string merupakan kumpulan dari beberapa karakter-karakter. Untuk membedakan string dengan karakter, dalam C++ dibedakan penulisannya. Suatu nilai merupakan string apabila diapit dengan tanda petik ganda "...", misalnya "SAYA". Sedangkan karakter (char) diapit dengan tanda petik tunggal, misal 's'.

Lantas bagaimana dengan "s"?? Dalam hal ini "s" juga merupakan string, meskipun karakter penyusunnya terlihat hanya satu. Akan tetapi pada kenyataannya, "s" disusun tidak hanya karakter 's' saja, melainkan terdapat pula karakter NULL atau '\0', yang berfungsi sebagai tanda akhir dari string.

Untuk mendeklarasikan suatu variabel merupakan string, maka perintahnya:

```
char variabel[maks karakter];
```

contoh:

```
char teks[20];
```

Perintah di atas bermakna bahwa teks merupakan variabel string dengan jumlah karakter yang dapat disimpan maksimal adalah 20 (sudah termasuk karakter NULL).

A. Inisialisasi String

Misalkan suatu variabel string katakanlah kalimat[30] akan diberi nilai "SAYA BELAJAR C++", maka perintahnya:

```
char kalimat[30] = "SAYA BELAJAR C++";
```

Program lengkapnya sbb:

```
#include <iostream.h>
#include <conio.h>

void main()
{
      int a;
      a = 20;

      char kalimat[30] = "SAYA BELAJAR C++";

      cout << "Nilai a = " <<  a << endl;
      cout << "Nilai kalimat = " << kalimat << endl;
}
```

Tidak boleh seperti di bawah ini!!

```
void main()
{
      int a;
      char kalimat[30];
      a = 20;

      kalimat = "SAYA BELAJAR C++";     // atau
      kalimat[30] = "SAYA BELAJAR C++";

      cout << "Nilai a = " <<  a << endl;
      cout << "Nilai kalimat = " << kalimat << endl;
}
```

C. Membaca String dari Keyboard

Selanjutnya bagaimana cara membaca string yang berasal dari keyboard?

Berikut ini contohnya:

```
#include<iostream.h>
#include<conio.h>

void main()
{
      char nama[20];
      char alamat[30];

      cout << "Masukkan nama Anda : ";
      cin.getline(nama, sizeof(nama));

      cout << "Masukkan alamat Anda : ";
      cin.getline(alamat, sizeof(alamat));

      cout << "Nama Anda : " << nama << endl;
      cout << "Alamat Anda : " << alamat << endl;
}
```

D. Mengcopy String

Kemudian bagaimana untuk mengassign suatu string dari variabel ke variabel lain? Misalnya **kata1** diberi string "HALLO". Selanjutnya **kata2** akan diberi string dari **kata1**. Untuk melakukan hal ini, **Anda tidak bisa memberikan perintah**

```
kata2 = kata1;
```

Perintah yang digunakan untuk keperluan di atas adalah dengan:

```
strcpy(kata2, kata1);      // mengcopy isi dari kata1 ke kata2
```

Contoh:

```
#include<iostream.h>
#include<conio.h>
#include<string.h>

void main()
{
      char kata1[20] = "HALLO";
      char kata2[20];

      strcpy(kata2, kata1);
      cout << "Kata1: " << kata1 << endl;
      cout << "Kata2: " << kata2 << endl;
}
```

E. Function untuk Operasi String

Function-function berikut ini dapat digunakan untuk memanipulasi string. Sebelum function digunakan, tambahkan file header **string.h** pada include.

- **Mengetahui panjang string dengan `strlen()`**

  sintaks:

  ```
  strlen(string)
  ```

  akan mereturn bilangan bulat yang menyatakan panjang string.

  Contoh:

  ```
  int panjangteks;
  char kalimat[30] = "BELAJAR C++ TIDAKLAH SULIT";

  panjangteks = strlen(kalimat);

  cout << "Panjang string : " << panjangteks;
  ```

- **Menggabungkan string dengan `strcat()`**

  Sintaks:

  ```
  strcat(string1, string2)
  ```

  menambahkan string2 ke string1.

  Contoh:

  ```
  char kata1[5] = "SATU ";
  char kata2[5] = "DUA";

  strcat(kata1, kata2);    // nilai kata1 menjadi "SATU DUA"
  ```

- **Mengkonversi ke huruf kapital dengan `strupr()`**

  Sintaks:

  ```
  strupr(string)
  ```

  Mengubah huruf kecil dari string ke huruf kapital.

  Contoh;

  ```
  char string1[30] = "aBcDefgHIJKLmno";

  strupr(string1);   //nilai string1 menjadi "ABCDEFGHIJKLMNO"
  ```

- **Mengkonversi ke huruf kecil dengan `strlwr()`**

  Sintaks:

  ```
  strlwr(string)
  ```

  Function ini kebalikan dari strupr().

- **Mencari Substring dengan `strstr()`**

  Misalkan diberikan suatu string "JAKARTA KOTA METROPOLITAN". Apakah string "METRO" terdapat dalam string tersebut?

  Untuk mengetahui hal ini dengan C++, kita dapat menggunakan function strstr().

  Sintaks:

  ```
  strstr(string1, string2);
  ```

  Function tersebut akan mereturn nilai 1 jika string2 merupakan substring dari string1, dan akan mereturn 0 jika tidak.

  Contoh:

  ```
  if (strstr("JAKARTA KOTA METROPOLITAN", "METRO") == 1)
     cout << "Merupakan substring";
  else cout << "Bukan merupakan substring";
  ```

- **Membalik string dengan `strrev()`**

  Bagaimana cara membalik string "C++" supaya diperoleh "++C"?
  Berikut ini perintah dalam C++,

  sintaks:

  ```
  strrev(string);
  ```

Contoh:

```
char kata[10] = “C++”;

strrev(kata);
cout << kata;
```